

IN THE CLAIMS

What is claimed is:

1. An algorithm to update a memory structure within a data plane of a forwarding device when one or more route updates occur in a control plane of the forwarding device, and wherein the algorithm is selected from one of the following algorithms:
 - a Basic Incremental Split-Merge (BISM) algorithm;
 - a Lazy Incremental Split-Merge (LISM) algorithm; and
 - a Down-support Split-Merge (DSM) algorithm.
2. A forwarding device having a computer readable storage medium adapted to update a memory structure within a data plane of the forwarding device when one or more route updates occur in a control plane of the forwarding device, wherein the forwarding device comprises:
 - the computer readable storage medium; and
 - an execution unit coupled to the computer readable storage medium and operable to fetch a sequence of instructions from the storage medium selected from a list of instructions for carrying out a Basic Incremental Split-Merge (BISM) operation, a Lazy Incremental Split-Merge (LISM) operation, and a Down-support Split-Merge (DSM) operation and updating the memory structure as a result thereof.
3. The forwarding device of claim 2, wherein the memory structure comprises a copy of a forwarding database stored within the computer readable storage medium.
4. The forwarding device of claim 3, wherein the execution unit comprises a central processor arranged within the control plane of the forwarding device, wherein the central processor is coupled to the computer readable storage medium for receiving the sequence of instructions corresponding to the selected operation, and upon execution of those instructions, for updating only the portion of the forwarding database affected by the route updates.

5. The forwarding device of claim 4, further comprising a local processor arranged within the data plane of the forwarding device, wherein the local processor is coupled to the central processor for receiving the updated portion of the forwarding database, and for using the updated portion to update the copy of the forwarding database stored within the memory structure.
6. The forwarding device of claim 5, wherein the computer readable storage medium, the central processor, the local processor and the memory structure are included within one or more line cards of the forwarding device.
7. The forwarding device of claim 6, wherein the forwarding device is selected from a group of network devices comprising: a router, a switch, a bridge or a gateway.
8. A method for updating a forwarding database, comprising:
 - forming a hierarchical tree structure of the forwarding database by splitting N number of prefixes within the database into a number of sub-databases bounded by N/T and $2N/T+1$, wherein each sub-database has no more than T number of prefixes, with T being less than N;
 - modifying the hierarchical tree structure in accordance with one or more update operations; and
 - updating a portion of the forwarding database to reflect modifications made to the hierarchical tree structure, wherein the updated portion corresponds to only those sub-databases affected by the update operations.
9. The method of claim 8, wherein said forming comprises, beginning with the most significant bit of the N number of prefixes, repeatedly splitting the N number of prefixes into a plurality of nodes extending between and including a root node and a plurality of leaf nodes, and wherein each of the plurality of leaf nodes corresponds to one of the sub-databases.

10. The method of claim 9, wherein said modifying comprises performing the update operations on one or more of the plurality of leaf nodes, wherein the update operations are selected from a group comprising: adding a new prefix to the forwarding database, deleting an existing prefix from the forwarding database and modifying an existing prefix in the forwarding database.

11. The method of claim 10, wherein said modifying comprises no further steps.

12. The method of claim 10, wherein said modifying further comprises one or more of the following steps:

splitting a leaf node, which has been modified to include more than T number of prefixes, into at least one additional pair of leaf nodes, each having less than T number of prefixes; and

merging a leaf node, which has been modified or split to include fewer than a minimum number of prefixes, with a parent node arranged closer to the root node than the leaf node having fewer than the minimum number of prefixes.

13. The method of claim 12, wherein said merging is performed only if:

the total number of nodes in the hierarchical tree structure is equal to or greater than $2N/T+1$; or

the total number of nodes in the hierarchical tree structure falls within a predetermined range of values immediately preceding and/or encompassing the value represented by $2N/T+1$; or

a predetermined time period has passed, in which no merging was performed.

15. The method of claim 13, wherein said merging further comprises repeatedly merging the leaf node and the parent node up towards the root node, if the number of prefixes within the leaf node, the parent node and any subsequently merged parent nodes remains less than the minimum number of prefixes.

16. The method of claim 12, wherein said merging is performed only if no other node exists below the parent node that can be paired with the leaf node, such that the combined number of prefixes within the leaf node and the other node is greater than T.
17. The method of claim 16, wherein said merging is performed no more than one time.
18. A lookup table stored in a computer-readable storage medium and construed in accordance with the method as recited in claim 8.
19. A computer or application specific integrated circuit (ASIC) residing within a forwarding device, a line card of an input/output port of the forwarding device, or a switch fabric of the forwarding device, for executing the method as recited in claim 8.
20. A computer readable storage medium, comprising:
 - a forwarding database comprising N number of prefixes split among a plurality of sub-databases, wherein each sub-database initially includes less than T number of prefixes, with T being less than N; and
 - an updating program that, when executed upon a processor:
 - (1) modifies a hierarchical tree structure in accordance with one or more update operations, wherein prior to execution of the updating program, the hierarchical tree structure included a number of branches extending from a root node to a plurality of leaf nodes, and wherein each of the plurality of leaf nodes corresponds to one of the plurality of sub-databases; and
 - (2) updates a portion of the forwarding database to reflect modifications made to the hierarchical tree structure, wherein the updated portion corresponds to only those sub-databases affected by the update operations.

21. The computer readable storage medium of claim 20, wherein the computer readable storage medium is directly coupled to, or incorporated within, the processor, and wherein at least a portion of the sub-database at each leaf node is contained within respective portions of the computer readable storage medium.
22. The computer readable storage medium of claim 21, wherein the computer readable storage medium comprises random access memory (DRAM or SRAM).
23. The computer readable storage medium of claim 21, wherein the updating program is stored within the computer readable storage medium, along with the forwarding database, or within a memory structure indirectly coupled to the processor.
24. The computer readable storage medium of claim 23, wherein a copy of the forwarding database is stored within the memory structure.
25. The computer readable storage medium of claim 21, wherein the memory structure comprises one or more of a random access memory (DRAM or SRAM), a content-addressable memory (CAM or TCAM), or a network search engine (NSE).